DTU Compute
Department of Applied Mathematics and Computer Science

Open Domain Q&A

Caroline Amalie Fuglsang-Damgaard (s164175)[1], Amalie Due Jensen (s160503)[1], Ida Riis Jensen (s161777)[1], Elisabeth Zinck (s164204)[1]

[1] Technical University of Denmark, Department of Applied Mathematics and Computer Science

Introduction

Open Domain Question Answering is the task of answering questions based on a large collection of documents. Such a task can be simplified to a two-stage framework: (1) A retriever which selects a small subset of documents (2) a reader that reads the retrieved documents and identifies the correct answer. This project focuses on the first stage: The retriever. More specifically, the goal is to re-implement Dense Passage Retrieval (DPR) using the power of Transformers to make a dense representation of the documents [4].

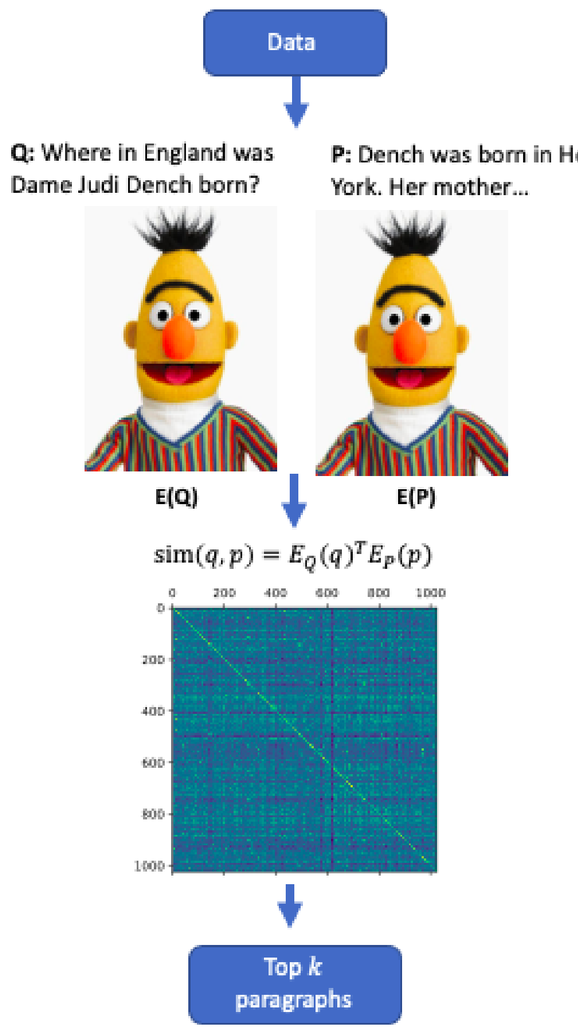

Figure 1: Dense Passage Retrieval system used in this project.

Methods

DPR

The DPR (illustrated in Figure 1) was trained and evaluated on the data set TriviaQA [1], using the Q&A pairs containing evidence documents from Wikipedia. Each document were split into 100-word paragraphs. The paragraph with the highest similarity using TF-IDF encoding with the question containing the answer was selected as the *positive paragraph*. A BERT model[3] was used to generate dense representations of questions and paragraphs. The inner product of the dense representations is the similarity, and the k paragraphs with highest similarity are returned for each question. The loss function used is the negative log likelihood of a positive passage p_i^+: $L(q_i, p_i^+, p_{i,1}^-, \cdots, p_{i,n}^-) = -\log \frac{e^{\mathrm{sim}\left(q_i, p_i^+\right)}}{e^{\mathrm{sim}\left(q_i, p_i^+\right)} + \sum_{j=1}^{n} e^{\mathrm{sim}\left(q_i, p_{i,j}^-\right)}}$ where $\mathrm{sim}(q_i, \cdot)$ is the similarity between question i and the positive or a negative passage from the batch, i.e. p_i^+ or $p_{i,j}^-$. Figure 1 illustrates how DPR was set up for this project.

BERT

BERT is a Deep Neural Network used for varying NLP tasks. The model processes entire sentences at once and the model learns language and context via transformers. Figures 2 and 3 shows the structure of $\mathrm{BERT}_{\mathrm{base}}$ as well as the special input scheme to the model, which introduces a classifier token ([CLS]) in front of each sentence.

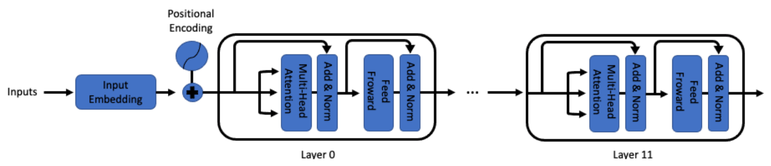

Figure 2: Model Architecture BERT base.

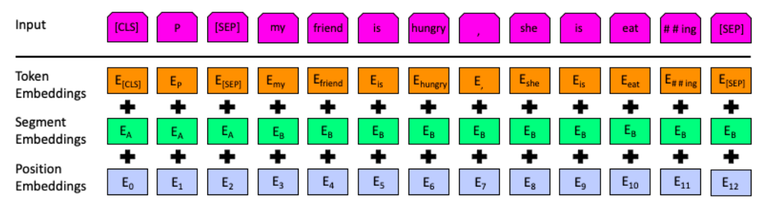

Figure 3: The input scheme for BERT.

During pre-training BERT learns to understand language and context using Masked Language Models and Next Sentence Prediction. $\mathrm{BERT}_{\mathrm{base}}$ is already pre-trained and can further be fine-tuned to solve specific NLP tasks. In this project the dense representation of a paragraph is the last hidden state of the [CLS] token and fine-tuning was done using the loss presented in the DPR section.

Transformers/Attention heads

Multi-Head attention (MHA) allows the non-sequential model to retrieve knowledge from different representation subspaces at different positions. The input to an MHA block is 3 vectors: q (queries), k (keys), and corresponding v (values). MHA applies h parallel attention layers, and the full mechanism is shown in Figure 4. MHA is calculated as $\mathrm{Attention}(Q, K, V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$.

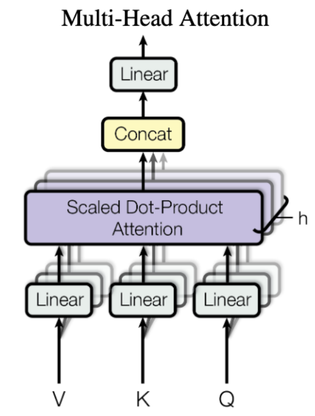

Figure 4: Reference: [2]. The inputs are sets of v, k, and q packed into matrices, V, K, and Q.

Results

Five different retrievers were implemented: random retrieval, TF-IDF representation, BERT base with no fine-tuning, BERT base with fine-tuned 11th layer, and finally BERT base with fine-tuned 9-11th layer, where the CLS token from the last 3 hidden layers were concatenated and used as output. The training was performed with batch_size = 16, 4 epochs, and learning rate = 5e-5. Figure 5 shows the accuracy of the retrievers for 1024 questions. With no fine-tuning, a simple TF-IDF retriever outperforms the DPR. When fine tuning the last layer (layer 11), the accuracy increases dramatically, and fine-tuning + concatenating from the 9-11th layer increases the performance even further.

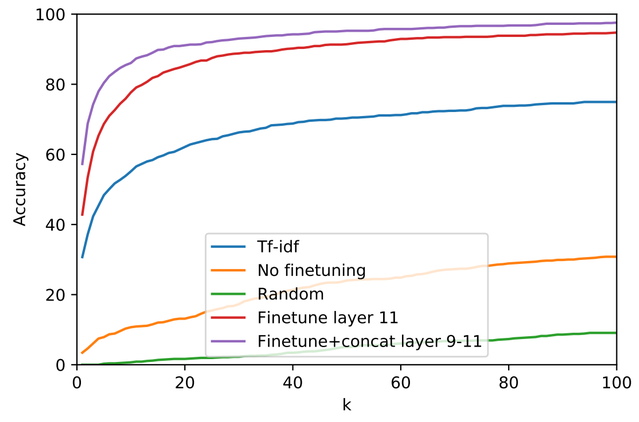

Figure 5: Accuracy of the retrievers as a function of the number of paragraphs returned.

Discussion

The dense representation of the data outperforms the sparse using TF-IDF. The high accuracy is, however, based on only 1024 pairs of questions and paragraphs, and the performance on larger data sets is unknown which makes comparison with other results difficult. More experimentation with learning rates, number of epochs, and the batch size, and using 'BERT-large-uncased' could potentially improve the results further. The accuracy is upwards biased because the positive passages do not represent the ground truth, but are only best guesses on the correct paragraphs, and the implementation of a reader is needed to fully evaluate the performance.

Conclusion

The results show that DPR using a fine tuned BERT base performed better than TF-IDF. More specific, the fine tuning and concatenation of 9-11th hidden layer resulted in an accuracy of $\approx$ 90% when returning k = 20 passages from 1024 question-answer pairs. This implies that the re-implementation of DPR using BERT was a success!

References

[1] M. Joshi et al. "triviaqa: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension". In: (2017).

[2] A. Vaswani et al. "Attention is all you need". In: *Advances in Neural Information Processing Systems* 2017-December (2017).

[3] J. Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: (2018).

[4] V. Karpukhin et al. "Dense Passage Retrieval for Open-Domain Question Answering". In: (2020).